13. ročník
Žilina 1. – 3. 7. 2025
Otvorený softvér vo vzdelávaní, výskume a v IT riešeniach
ossconf.fri.uniza.sk
OSSConf

OSSConf 2025: 1–7

# O jedné mapě na zakázku, část 2: Počet obarvení $n$ barvami

PAVEL STŘÍŽ (CZ)

**Abstrakt.** Článek počítá počet $n$-obarvení v daném grafu, chceme-li mapě krajů či okresů. Vedle běžných 4 barev užitých v teorému čtyř barev jsem dopočítal $n$-obarvení pro přípustné barvy. Dvacetidvouuzlový graf byl stále ještě výpočetně přijatelný.

**Klíčová slova.** Optimalizace, R, Sage, Python, Picat, OpenStreetMap.

## ABOUT PREPARATION OF ONE MAP, PART 2: THE NUMBER OF $N$-COLOURINGS

**Abstract.** The article computes number of $n$-colourings in a given map/graph. Next to four colours used in the Four Colour Theorem, I tried to use all possible number of colours and compute its number of $n$-colourings. 22-node graph was still computable.

**Keywords.** Optimization, R, Sage, Python, Picat, OpenStreetMap.

## 1. Zpět k R

Rovnou naváži na první část. Při experimentech s programy pro lineární programování kolem výpočtů v sudoku mně vrtalo hlavou, jak v R nechat spočítat model ILP. Knihovna `lpsolve` mě moc nezaujala, při rozhození modelu do sloupců se člověk začne ztrácet. Načtení .lp souboru je závislé od jeho formátu. Hledal jsem proto jinou cestu, když mi to na první ani druhou dobrou nejelo.

Masívně užívám program Picat, `https://picat-lang.org/`, kde se lze mezi MIP, SAT, SMT a CP přepínat. V první části tohoto článku je ukázka generování přes Python. Přepínám se z `import sat` na `import mip`. V pozadí se uloží soubor `__tmp.lp`. Ten ještě R neumí zpracovat, resp. jsem cestu nenašel, použil jsem následující udělátko.

V Pythonu přes `pyscipopt` přeukládám do formátu FreeMPS.

```
from pyscipopt import Model
scip = Model()
scip.readProblem(filename="__tmp.lp")
scip.writeProblem(filename="pokonverzi.mps",
   trans=False, genericnames=False)
```

Takový soubor si lze v R načíst a například přes NEOS server nechat spočítat.

```
library("ROI")
library(ROI.plugin.lpsolve)
```

```
library(ROI.plugin.neos)
(x <- ROI.plugin.lpsolve::read.lp("pokonverzi.mps", type="freemps"))
opt <- ROI_solve(x, solver="neos", method="scip", email="Email-na-NEOSu")
opt; solution(opt)
```

Přebereme výsledky a vykreslíme přes `ggplot2`. Naznačím jádro. Pracovně jsem si okresy zpracoval ze surových dat v bočním souboru. Z výstupů jsem si vybral potřebné údaje a nechal vykreslit. Grafický náhled viz první část tohoto příspěvku.

```
library(ggredist)
library(ggplot2)
library(sf)
source("stefanovi-data.R")
mal <- st_sf(geom=malseznam)
malshort<-solution(opt)[2:150]
ggplot(mal, aes(fill = malshort))+geom_sf()+theme_map()
```

## 2. Zpět k Pythonu: případ Sage

Podobného vykreslení ze surových dat lze dosáhnout i v Pythonu, např. přes knihovnu `matplotlib`. Jen naznačím strukturu.

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.patches import Polygon
combined=Polygon([ (0,1), (1,2), (2,1), (1,0) ])
fig, ax = plt.subplots()
ax.add_patch(combined)
plt.ylim(0,3)
plt.xlim(0,3)
plt.show()
```

Poněvadž jsem se nedopracoval výsledku přes knihovnu `networkx`, hledal jsem další cesty. Zkusil jsem si SageMathCell, `https://sagecell.sagemath.org/`, či `https://cocalc.com/`. Řada věcí je placených, ale na testy to stačilo. Za pozornost stojí možnost týmové práce, diskuze i videohovoru. Náš problém spadá v dokumentaci do této části, viz `https://doc.sagemath.org/html/en/reference/graphs/sage/graphs/graph_coloring.html`.

```
from sage.graphs.graph_coloring import number_of_n_colorings
from sage.graphs.graph_plot import GraphPlot
# Jihomoravský kraj
G = Graph({
  17: [67,68],
  32: [61,68],
  49: [61,67],
  61: [32,49,66,67,68],
  66: [61],
```

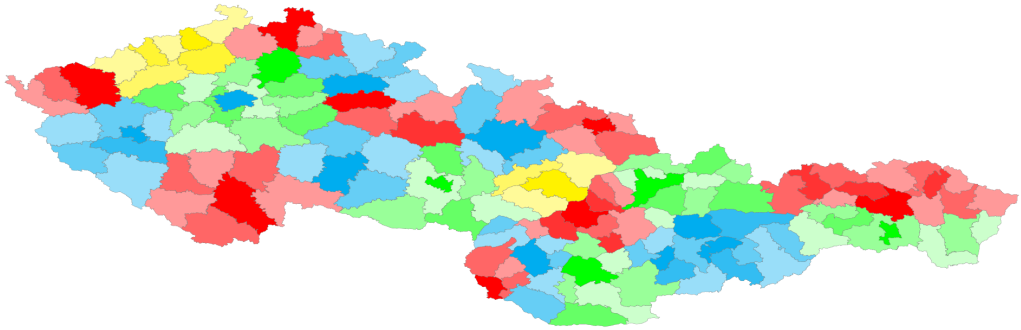
```
  67: [17,49,61,68],
  68: [17,32,61,67],
})
for n in range(8): # [4]
  print(number_of_n_colorings(G,n))
G.graphplot(layout='graphviz', prog="twopi").show()
```

A právě během toho testu jsem si zkusil čtyři barvy, ale pak i jiný počet barev. A dostal jsem nápad propočítat všechny možnosti pro okresy a kraje. U krajů je největší počet uzlů (22), což byla výpočetní výzva. Sage začal kolabovat u šesti barev. Dostupných 45 minut na výpočty by určitě nestačilo. Nestačil by na to v rozumném čase ani jeden počítač, bylo potřeba výpočty distribuovat a podívat se po nějaké úspoře, spisovně po lepším algoritmu.

## 3. Zpět k rekurzi

Od dvojiček, Rudolfa Blaška a Aleše Kozubíka, prakticky ve stejnou chvíli, došla prosba, jestli by nebylo možné vybarvit kraje (hlavní barva) a poté vybarvit okresy uvnitř kraje (odstín hlavní barvy). Na vše požadavek, aby zkoumaní sousedi neměli stejnou barvu. Poněvadž jsou počty krajů a okresů v řádech desítek, mohli jsme zkusit až 6 barev.

Při nejnižším možném počtu barev jsem vybral první obarvení a užil. Jako bonus se krajské město (vyjma Středočeského kraje) nastavilo nejsytější barvou. Zde je závěrečná ukázka. Kdyby se šlo na úroveň ohraničení měst, dokáži si představit užití šrafování v dané barvě a intenzitě.



Druhou věc, kterou jsem si zkusil bylo vygenerovat počty různých obarvení při daném počtu barev. Obvody Prahy a okresy Bratislavy a Košic jsem si sloučil dohromady.

Co mě během bádání zaujalo je, že můžeme použít menší počet barev než 4, 4 barvami lze oblasti vždy vybarvit, ale nic nás neomezuje v tom užít i víc barev. Matematická teorie hovoří o tom, že pokud je víc barev než oblastí, počet

obarvení je nula, angl. number of $n$-colourings. Lze si to představit tak, že obarvíme všechny oblasti, ale když nám některé barvy zůstanou, pak to není obarvení přesně $n$ barvami.

Při obarvování můžeme barvy prohazovat (matematik Štefan Peško upřesnil, že se jedná o automorfismus), pokud tedy najdeme počet neautomorfních obarvení násobíme zjištěný počet faktoriálem počtu barev. Výsledky jsou zmíněny v následujících dvou tabulkách. V první je počet neautomorfních obarvení, ve druhé pak celkový počet obarvení.

| **Zkoumaná oblast** | **Uzlů** | **Min.** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hlavní město Praha | 1 | 1 | 1 | – | – | – | – | – | – | – | – | – | – | – | – |
| Středočeský kraj | 12 | 4 | – | – | – | 230 | 5320 | 22547 | 31283 | 18254 | 5036 | 681 | 43 | 1 | – |
| Jihočeský kraj | 7 | 3 | – | – | 1 | 15 | 25 | 10 | 1 | – | – | – | – | – | – |
| Plzeňský kraj | 7 | 4 | – | – | – | 8 | 19 | 9 | 1 | – | – | – | – | – | – |
| Karlovarský kraj | 3 | 3 | – | – | 1 | – | – | – | – | – | – | – | – | – | – |
| Ústecký kraj | 7 | 3 | – | – | 1 | 15 | 25 | 10 | 1 | – | – | – | – | – | – |
| Liberecký kraj | 4 | 3 | – | – | 2 | 1 | – | – | – | – | – | – | – | – | – |
| Královéhradecký kraj | 5 | 3 | – | – | 1 | 3 | 1 | – | – | – | – | – | – | – | – |
| Pardubický kraj | 4 | 3 | – | – | 1 | 1 | – | – | – | – | – | – | – | – | – |
| Kraj Vysočina | 5 | 3 | – | – | 1 | 3 | 1 | – | – | – | – | – | – | – | – |
| Jihomoravský kraj | 7 | 3 | – | – | 2 | 22 | 31 | 11 | 1 | – | – | – | – | – | – |
| Olomoucký kraj | 5 | 3 | – | – | 4 | 5 | 1 | – | – | – | – | – | – | – | – |
| Moravskoslezský kraj | 6 | 3 | – | – | 2 | 7 | 6 | 1 | – | – | – | – | – | – | – |
| Zlínský kraj | 4 | 3 | – | – | 1 | 1 | – | – | – | – | – | – | – | – | – |
| Bratislavský kraj | 4 | 3 | – | – | 1 | 1 | – | – | – | – | – | – | – | – | – |
| Trnavský kraj | 7 | 3 | – | – | 8 | 46 | 46 | 13 | 1 | – | – | – | – | – | – |
| Trenčínský kraj | 9 | 3 | – | – | 4 | 140 | 506 | 490 | 171 | 23 | 1 | – | – | – | – |
| Nitranský kraj | 7 | 3 | – | – | 1 | 15 | 25 | 10 | 1 | – | – | – | – | – | – |
| Žilinský kraj | 11 | 4 | – | – | – | 252 | 3024 | 7770 | 6951 | 2646 | 462 | 36 | 1 | – | – |
| Banskobystrický kraj | 13 | 4 | – | – | – | 300 | 12273 | 77681 | 152670 | 124688 | 48877 | 9837 | 1026 | 52 | 1 |
| Prešovský kraj | 13 | 3 | – | – | 2 | 1150 | 29467 | 147451 | 247780 | 179753 | 64015 | 11881 | 1155 | 55 | 1 |
| Košický kraj | 8 | 3 | – | – | 16 | 146 | 230 | 111 | 19 | 1 | – | – | – | – | – |

| **Zkoumaná oblast** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hlavní město Praha | 1 | – | – | – | – | – | – | – | – | – | – | – | – |
| Středočeský kraj | – | – | – | 5520 | 638400 | 16233840 | 157666320 | 736001280 | 1827463680 | 2471212800 | 1716422400 | 479001600 | – |
| Jihočeský kraj | – | – | 6 | 360 | 3000 | 7200 | 5040 | – | – | – | – | – | – |
| Plzeňský kraj | – | – | – | 192 | 2280 | 6480 | 5040 | – | – | – | – | – | – |
| Karlovarský kraj | – | – | 6 | – | – | – | – | – | – | – | – | – | – |
| Ústecký kraj | – | – | 6 | 360 | 3000 | 7200 | 5040 | – | – | – | – | – | – |
| Liberecký kraj | – | – | 12 | 24 | – | – | – | – | – | – | – | – | – |
| Královéhradecký kraj | – | – | 6 | 72 | 120 | – | – | – | – | – | – | – | – |
| Pardubický kraj | – | – | 6 | 24 | – | – | – | – | – | – | – | – | – |
| Kraj Vysočina | – | – | 6 | 72 | 120 | – | – | – | – | – | – | – | – |
| Jihomoravský kraj | – | – | 12 | 528 | 3720 | 7920 | 5040 | – | – | – | – | – | – |
| Olomoucký kraj | – | – | 24 | 120 | 120 | – | – | – | – | – | – | – | – |
| Moravskoslezský kraj | – | – | 12 | 168 | 720 | 720 | – | – | – | – | – | – | – |
| Zlínský kraj | – | – | 6 | 24 | – | – | – | – | – | – | – | – | – |
| Bratislavský kraj | – | – | 6 | 24 | – | – | – | – | – | – | – | – | – |
| Trnavský kraj | – | – | 48 | 1104 | 5520 | 9360 | 5040 | – | – | – | – | – | – |
| Trenčínský kraj | – | – | 24 | 3360 | 60720 | 352800 | 861840 | 927360 | 362880 | – | – | – | – |
| Nitranský kraj | – | – | 6 | 360 | 3000 | 7200 | 5040 | – | – | – | – | – | – |
| Žilinský kraj | – | – | – | 6048 | 362880 | 5594400 | 35033040 | 106686720 | 167650560 | 130636800 | 39916800 | – | – |
| Banskobystrický kraj | – | – | – | 7200 | 1472760 | 55930320 | 769456800 | 5027420160 | 17736485760 | 35696505600 | 40954636800 | 24908083200 | 6227020800 |
| Prešovský kraj | – | – | 12 | 27600 | 3536040 | 106164720 | 1248811200 | 7247640960 | 23229763200 | 43113772800 | 46103904000 | 26345088000 | 6227020800 |
| Košický kraj | – | – | 96 | 3504 | 27600 | 79920 | 95760 | 40320 | – | – | – | – | – |

Zvláštní kategorii tvoří kraje, kde vychází obrovský počet obarvení, proto jsem je z okresů vyčlenil do samostatné tabulky.

| Barev | Neautomorfní p.o. | Celkový počet obarvení | Počet cifer |
|---|---|---|---|
| 0–3 | 0 | 0 | 1 |
| 4 | 241056 | 5785344 | 7 |
| 5 | 334406880 | 40128825600 | 11 |
| 6 | 29332429392 | 21119349162240 | 14 |
| 7 | 525980776992 | 2650943116039680 | 16 |
| 8 | 3204670864848 | 129212329270671360 | 18 |
| 9 | 8679321523496 | 3149552194446228480 | 19 |
| 10 | 12261482221996 | 44494466687179084800 | 20 |
| 11 | 10013611391758 | 399711323202525734400 | 21 |
| 12 | 5066436397528 | 2426831140714148044800 | 22 |
| 13 | 1665710385805 | 10372413219183759744000 | 23 |
| 14 | 367671148358 | 3205294243739212584960 0 | 23 |
| 15 | 55646946232 | 72768085245060581376000 | 23 |
| 16 | 5839858189 | 122186125864163192832000 | 24 |
| 17 | 425574024 | 151371330061025378304000 | 24 |
| 18 | 21326264 | 136538711875013640192000 | 24 |
| 19 | 716547 | 87164431762647343104000 | 23 |
| 20 | 15336 | 37310985197396951040000 | 23 |
| 21 | 188 | 9605097128281374720000 | 22 |
| 22 | 1 | 1124000727777607680000 | 22 |

## 4. Technické pozadí výpočtů

Jak však tato obarvení pro tak velký počet uzlů najít v rekurzi s komplikovanou stromovou strukturou?

V matematice, speciálně v teorii grup a kombinatorice, se hledají symetrie a podobné vztahy, které by výpočty usnadnily. Zde se mi podařilo v rekurzi stromovou strukturu obarvení uzlů snížit na hledání jen neautomorfních větví. Pracovně to nazývám nalézt tzv. „prvoobarvení“. Je to rostoucí sekvence prvních výskytů barev, kdy každá barva je zastoupena alespoň jednou. Počet dalších výskytů barev nesmí překročit hranici počet uzlů mínus počet barev, takovou větev jsem v rekurzi vyřadil.

Zmíním několik ukázek a zdůvodnění vyřazení větve z dalšího zkoumání v rekurzi. 4 barvy, 7 uzlů, graf nechť je smyšlený.

| | |
|---|---|
| B | Na prvním místě nemůže být druhá barva v pořadí, nedostali bychom uspořádanou $n$-tici. Podobně by se vyřadily C a D. |
| ABABD | Na páté pozici mohou být barvy A až C, protože nejvyšší předchozí je barva B. Lze připustit jen o jednu vyšší pozici (barva C). |

| | |
|---|---|
| ABABCB | Pokud by se jednalo o vybarvení 4 barvami, chybí na poslední pozici barva D. Pokud rekurze ještě běží, D lze užít, žádný ze sousedů barvu D nemá. |
| AAABBBC | Vyplnilo se sice 7 uzlů, ale neužily se všechny barvy (D). |
| ABBABA | Zbývá vybarvit poslední uzel, ale potřebujeme užít ještě barvy C a D, na ty není už dost uzlů. Volné uzly jsou 3. až 6., tedy 4 uzly, ale mohou být jen tři (7 uzlů mínus 4 barvy je maximum). |
| ABCDBCD | Tato situace by v závislosti na grafu mohla být platným obarvením, ale prvoobarvení by skončilo už při ABCD. |
| DCBA | Toto obarvení se získá z nejkratšího možného prvoobarvení ABCD permutací, počet nalezených obarvení se násobí 4!. Technicky měla být větev vyřazena již na první pozici u barvy D. |

Zde je ukázka prvoobarvení v plné délce pro 22 uzlů krajů a 4 barvy. Každá sekvence začíná první barvou (A) a končí poslední barvou (D). Barva vždy splňuje podmínku, že je někde v rozsahu od A až nejvyšší barva z levých uzlů + 1 pozice. Podtržením naznačuji první výskyt dané barvy. V této malé ukázce je 15 prvoobarvení. Více jich není.

| **22 krajů ve 4 barvách** | | **Počet obarvení ve větvi** | |
|---|---|---|---|
| **Pořadí** | **Prvoobarvení** | **neautomorfní** | **celkem, krát 4!** |
| 1 | A B A C A D | 22032 | 528768 |
| 2 | A B A C B A C A C D | 1296 | 31104 |
| 3 | A B A C B A C A D | 1296 | 31104 |
| 4 | A B A C B A C D | 6480 | 155520 |
| 5 | A B A C B A D | 9072 | 217728 |
| 6 | A B A C B D | 22032 | 528768 |
| 7 | A B A C D | 18144 | 435456 |
| 8 | A B C A B C A C A D | 1296 | 31104 |
| 9 | A B C A B C A C D | 1296 | 31104 |
| 10 | A B C A B C A D | 6480 | 155520 |
| 11 | A B C A B C D | 9072 | 217728 |
| 12 | A B C A B D | 22032 | 528768 |
| 13 | A B C A C D | 22032 | 528768 |
| 14 | A B C A D | 18144 | 435456 |
| 15 | A B C D | 80352 | 1928448 |
| | **Součet** | **241056** | **5785344** |

Možná si kladete otázku, jaktože neexistuje sekvence typu A A A A A B C D. Je to dané zadáním úlohy, že sousedící uzly nesmí mít stejnou barvu. Uzly si můžeme různě přečíslovat (nemusí to být cesta ve stylu známém z teorie grafů),

ale podmínku sousedících uzlů nesmíme opominout. Například u Košického kraje z okresů nelze Eulerovu cestu, tzv. jednotahovku, ani najít. Obecně však taková sekvence může být platná, záleží na konkrétním grafu a užitém pořadí uzlů.

Poté hledáme platná obarvení z těchto sekvencí rekurzí od určité hloubky a jejich součet nám dá počet neautomorfních obarvení. To pak stačí vynásobit faktoriálem počtu barev. Samotná prvoobarvení nebyla na rozložení rekurze vhodná, některé sekvence byly spočítány rychle, jiné pomalu. Nápad tedy byl, vzít sice jen tato prvoobarvení, ale do určité hloubky. To se nakonec užilo. Post mortem řečeno: ještě lepší varianta by byla, aby se kratší prvoobarvení než nastavená hloubka rekurze nechaly rozvinout do nastavené hloubky. To by bylo ještě víc časově rovnoměrné. Třeba užiji v budoucnu.

Touto fintou se mi podařilo eliminovat větve s proházenými barvami v duchu kombinatoriky. Následně jsem rozhodil rekurzivní výpočty na několik stovek jader na cca tři dny. A bylo spočteno! V průměru jsem užil 17000 úloh na barvu, to bylo při hloubce 11 uzlů. Lepší bylo užít hloubku 9 či 10 uzlů, těch úloh bylo zbytečně moc.

Poznámka. Nejsem zběhlý v C++, zdrojové kódy v Pythonu, především užité datové struktury, jsem si nechal převádět v `https://products.codeporting.ai/convert/python-to-cpp`. Věřím tomu, že podobných nástrojů, ať už s nebo bez AI asistence, se rodí nespočet. Bez C++ to bylo nepočítatelné, Python není na výpočty optimalizován. Možná by zabral ještě Rust či Go, ale nezkoušel jsem je. Poctivě se učím C++. Dorazil mi tip, že Python nemám vzdávat, ale užít knihovny `numpy` a `pandas`. Zůstává mi otevřené s otazníkem.

## 5. Postřeh

Při výpisu počtu obarvení na uzlech v rekurzi to vypadá, že by se dalo uspořit ještě víc, protože počty na větvích na mnoha místech vycházely stejné. Chce se to však víc ponořit do teorie grafů, možná do kombinatoriky. Rekurzi se nejspíš vyhnout nedá, ale naděje na úsporu umírá poslední. Po objevení takového Svatého Grálu by se dal počet obarvení spočítat i pro větší grafy/mapy. Jak však velké je těžké odhadovat. Dávám si mezi otevřené problémy a důležité úkoly po OSSConf.

## Kontaktní adresa

**Ing. Pavel Stříž, Ph.D.,** Nakladatelství Martin Stříž, U Škol 940, Bučovice, okres Vyškov, 685 01, Česká republika,
*E-mailová adresa*: `pavel@striz.cz`